



TITLE:

# Factoring Multivariate Polynomial modulo a Monic Reducible Polynomial(Theory and applications in computer algebra)

AUTHOR(S):

北本, 卓也; 佐々木, 建昭

---

CITATION:

北本, 卓也 ...[et al]. Factoring Multivariate Polynomial modulo a Monic Reducible Polynomial(Theory and applications in computer algebra). 数理解析研究所講究録 1993, 848: 13-16

ISSUE DATE:

1993-09

URL:

<http://hdl.handle.net/2433/83666>

RIGHT:

## Factoring Multivariate Polynomial modulo a Monic Reducible Polynomial

筑波大学数学系 北本 卓也 (Takuya KITAMOTO)<sup>1)</sup>

” 佐々木 建昭 (Takeaki SASAKI)<sup>2)</sup>

**Abstract.** 本稿では、与えられた多項式  $F$  と  $D$  に対し、 $F \equiv GH \pmod{D}$  なる形の因数分解を扱う。ここで、 $F$  と  $D$  は 1 変数でも多変数でもよく、若干の条件を満たし、モニックであることを仮定するが、 $D$  は既約でなく可約である。もしも  $D$  が既約な 1 変数 (多変数) 多項式ならば、上記の分解は代数的数体 (代数関数体) 上の因数分解に他ならない。したがって、本稿で扱う因数分解は、従来の代数拡大体 (代数関数体) 上の因数分解を若干拡張したものである。

### 1. Notation

$K$  : 数体  
 $K[x, y, z_1, \dots, z_n]$  : 数体  $K$  上の多項式環  
 $F(x, y, z_1, \dots, z_n) \in K[x, y, z_1, \dots, z_n]$  : 因数分解されるべき多項式  
 $D(y, z_1, \dots, z_\nu) \in K[y, z_1, \dots, z_\nu]$  : 与えられた法,  $\nu \leq n$   
 $D = D_1^{k_1} D_2^{k_2} \dots D_m^{k_m}$  :  $D$  の  $K$  上での既約因数分解

ただし、 $D = D(y)$  あるいは  $D = D(y, z_1, \dots, z_\nu)$ ,  $\nu < n$ , でももちろん良い。

### 2. Algorithm

まず、 $F = F(x, y)$ ,  $D = D(y)$  の場合、つまり法  $D(y)$  が 1 変数の多項式として与えられた時を考える。アルゴリズムは次の 4 ステップからなる。

#### アルゴリズム 1

入力 ...  $F(x, y)$  : 因数分解されるべき多項式  
 $D(y)$  : 与えられた法  
出力 ...  $(\text{mod } D(y))$  上での  $F(x, y)$  の既約因数分解

1. 与えられた法  $D$  を  $K$  上で以下のように既約因数分解する。

$$D = D_1^{k_1} D_2^{k_2} \dots D_m^{k_m}$$

(一般性を失うことなく、 $k_1 \leq k_2 \leq \dots \leq k_m$  とする)

<sup>1)</sup>kita@math.tsukuba.ac.jp

<sup>2)</sup>sasaki@math.tsukuba.ac.jp

2.  $D$  の各既約因子  $D_i$  ( $i = 1, \dots, m$ ) を法として、代数的拡大体上の因数分解アルゴリズムを用い、 $F$  を以下のように既約因数分解する。

$$F \equiv F_{i,1} F_{i,2} \cdots F_{i,r_i} \pmod{D_i} \quad (1)$$

この時、これらの既約因子の中には同じものが現れることもある。

3. Step 2 で得られた因子  $F_{i,1}, F_{i,2}, \dots, F_{i,r_i}$  のいくつかを掛け合わせたものを  $F_i$  ( $i = 1, \dots, m$ ) とし、

$$F' \equiv F_1 \pmod{D_1}$$

$$F' \equiv F_2 \pmod{D_2}$$

...

$$F' \equiv F_m \pmod{D_m}$$

を満たす  $F'$  を中国人剰余定理により求める。

次に、 $F'$  による  $F$  の  $(\text{mod } D_1 D_2 \cdots D_m)$  上での試し割りを行なう。割り切れるものがあれば、それを  $F$  の既約因子とし、各  $i = 1, \dots, m$  に対し  $F_i$  に含まれる  $(\text{mod } D_i)$  での因子を除く。

この手順を  $F_{i,j}$  ( $j = 1, \dots, r_i$ ) のあらゆる組合せに対して行なう。

なお  $F$  はモニックであることより  $F_1, F_2, \dots, F_m$  もモニックとなるので、結局  $F'$  もモニックとなる。これより  $F'$  での試し割が可能ながことが保証される。

4. Step 3 で得られた既約因数分解を以下とする。

$$F \equiv F_1^{(0)} F_2^{(0)} \cdots F_l^{(0)} \pmod{D_1 D_2 \cdots D_m}$$

ここで上式の  $F_i^{(0)}$  ( $i = 1, \dots, l$ ) が互いに異なるならば (これが、abstract の ‘若干の条件’ に相当する)、これを拡張された Hensel 構成を用いて (拡張された Hensel 構成については後述する)

$$F \equiv F_1^{(k_m)} F_2^{(k_m)} \cdots F_l^{(k_m)} \pmod{D_1^{k_1} D_2^{k_2} \cdots D_m^{k_m}}$$

へとヘンゼルリフティングすることができる。 //

つぎに、 $F = F(x, y, z_1, \dots, z_n)$ ,  $D = D(y, z_1, \dots, z_\nu)$  つまり  $F$  が  $n+2$  変数の多項式、 $D$  が  $\nu+1$  変数の多項式として与えられた時を考える。この場合には、代数的関数体上の因数分解のアルゴリズム [2] を適用する。具体的には次のようになる。

## アルゴリズム 2

1.  $F$  および  $G$  の変数  $z_1, z_2, \dots, z_\nu$  に適当な数値を代入し、結果をそれぞれ  $F', D'$  とする。
2. 上記の  $F', D'$  に対し、アルゴリズム 1 を適用し、 $F'$  の  $(\text{mod } D')$  での既約因数分解を行なう。
3. ヘンゼル構成を用いて  $z_1, \dots, z_\nu$  を回復し、得られた因子 (理論的には無限級数因子となる) を組み合わせて、多項式因子を得る。 //

### 3. 拡張された Hensel 構成について

ここでは、Step 4 で使われている拡張された Hensel 構成について説明する。

まず、 $D$  を

$$D = \tilde{D}_1 \tilde{D}_2^2 \cdots \tilde{D}_{k_m}^{k_m}$$

と無平方分解する ( $\tilde{D}_i$  は  $D_i$  と異なることに注意。各  $\tilde{D}_i$  は無平方であるが既約であるとは限らない)。ここで、

$$\begin{aligned} D^{(1)} &= \tilde{D}_1 \tilde{D}_2 \cdots \tilde{D}_{k_m} \\ D^{(2)} &= \tilde{D}_2 \cdots \tilde{D}_{k_m} \\ &\vdots \\ D^{(k_m)} &= \tilde{D}_{k_m} \end{aligned}$$

とおく。  $D^{(1)} = D_1 D_2 \cdots D_m$  であるから、アルゴリズム 1 の Step 1 ~ 3 で

$$F \equiv F_1^{(0)} F_2^{(0)} \cdots F_l^{(0)} \pmod{D^{(1)}}$$

なる既約因数分解を求めることが出来る。

ここで、アルゴリズム 1 で述べたように、 $F_i^{(0)}$  ( $i = 1, \dots, n$ ) は互いに異なる、つまり上の既約因数分解は無平方であると仮定する。

次に  $\pmod{D^{(1)}}$  を

$$\pmod{D = D_1^{(1)} D_2^{(2)} \cdots D_{k_m}^{(k_m)}}$$

へとリフティングする。

上の仮定より  $F$  が  $\pmod{D^{(1)}}$  で無平方である、つまり

$$F \equiv F_1^{(0)} F_2^{(0)} \cdots F_l^{(0)} \pmod{D^{(1)}}$$

$$F_1^{(0)}, F_2^{(0)}, \dots, F_l^{(0)} \text{ は } \pmod{D^{(1)}} \text{ で既約かつ互いに異なる}$$

であるので、

$$F_i^{(1)} = F_i^{(0)} + \Delta F_i^{(1)} D^{(1)} \quad (i = 1, \dots, l)$$

とおき、

$$F \equiv F_1^{(1)} \cdots F_l^{(1)} \pmod{D^{(1)} D^{(2)}}$$

となるように、 $\Delta F_1^{(1)}, \dots, \Delta F_l^{(1)}$  を決める。そのためには

$$[F - F_1^{(0)} \cdots F_l^{(0)}] / D^{(1)} \equiv \Delta F_1^{(1)} F_2^{(0)} \cdots F_l^{(0)} + \cdots + \Delta F_l^{(1)} F_1^{(0)} \cdots F_{l-1}^{(0)} \pmod{D^{(2)}}$$

となるように決めれば良いが、 $F_1^{(0)}, \dots, F_l^{(0)}$  が互いに素ゆえ、これは可能である。次に、

$$F_i^{(2)} = F_i^{(0)} + \Delta F_i^{(1)} D^{(1)} + \Delta F_i^{(2)} D^{(1)} D^{(2)} \quad (i = 1, \dots, l)$$

とおき

$$[F - F_1^{(1)} \cdots F_l^{(1)}] / D^{(1)} D^{(2)} \equiv \Delta F_1^{(2)} F_2^{(0)} \cdots F_l^{(0)} + \cdots + \Delta F_l^{(2)} F_2^{(0)} \cdots F_{l-1}^{(0)} \pmod{D^{(3)}}$$

となるように決める。以下、これを繰り返せば良い。

以上の算法が示すように、一般の  $D$  の場合、その無平方成分  $D^{(1)}$  で  $F$  が無平方な因子に因数分解できれば、その因子をリフティングしたものがそのまま既約因子となる。

$F$  が  $\pmod{D^{(1)}}$  で無平方でない場合については、今後の課題である。

## 参 考 文 献

- [1] T. Sasaki, T. Saito and T. Hilano, "Analysis of Approximate Factorization Algorithm, I", Japan J. Indust. Appl. Math., Vol. 9, No. 3, October 1992.
- [2] T. Sasaki and M. Sasaki, "Unified Method for Multivariate Polynomial Factorizations", Japan J. Indust. Appl. Math., to appear.
- [3] B. M. Trager, "Algebraic Factoring and Rational Function Integration", Proc. SYMSAC '76, ACM, pp. 219-226, 1976.
- [4] P. J. Weinberger, L. P. Rothchild, "Factoring Polynomials Over Algebraic Number Fields", ACM Trans. Math. Software, Vol. 2, No. 4, pp. 335-350, 1976.